



Megacity – 424B

Link submit: <http://codeforces.com/problemset/problem/424/B>

Solution:

C++: <http://ideone.com/hGq7wX>

python: <https://ideone.com/Z1RSIw>

Java: <https://ideone.com/qWnvB>

Tóm tắt đề: Chính quyền thành phố Tomsk mong muốn mở rộng thành phố này để thành phố này trở thành siêu đô thị (Megacity).

Thành phố của Tomsk có thể được biểu diễn như là điểm trên mặt phẳng với tọa độ $(0,0)$. Thành phố được bao quanh với n điểm khác. Vị trí i có tọa độ (x_i, y_i) với dân số là k_i người. Bạn có thể mở rộng ranh giới của thành phố đến một vòng tròn có bán kính r . Trong trường hợp tất cả vị trí bên trong vòng tròn và trên biên của vào tròn được xem là bao gồm bên trong thành phố. Mục tiêu của bạn là viết một chương trình xác định bán kính tối thiểu, cần thiết để mở rộng ranh giới Tomsk để nó trở thành một siêu đô thị.

Input

Dòng đầu tiên chứa 2 số nguyên n và s ($1 \leq n \leq 10^3$; $1 \leq s < 10^6$) là số lượng vị trí xung quanh thành phố Tomsk và dân số của thành phố. Sau đó là n dòng theo sau. Dòng thứ i chứa 3 số nguyên x_i và y_i là tọa độ của vị trí thứ i , và số k_i là số người bên trong nó ($1 \leq k_i < 10^6$). Mỗi tọa độ là một số nguyên không quá 10^4 .

Dữ liệu luôn đảm bảo rằng không có 2 vị trí tại cùng một điểm.

Output

In ra "-1" nếu thành phố Tomsk không thể trở thành siêu đô thị, Ngược lại trong dòng đầu tiên in ra một con số thực, số tối thiểu bán kính của vòng tròn thành phố cần phải mở rộng để trở thành siêu đô thị.

Câu trả lời được xem là đúng nếu sai số tuyệt đối hay tương đối không vượt 10^6 .

4 999998 1 1 1 2 2 1 3 3 1 2 -2 1	2.8284271
---	-----------

Hướng dẫn giải:

Rõ ràng nếu như ta xem như mỗi người là một đỉnh trên một đồ thị, thì việc đánh số cho một người có số là 0, sau đó ta đi đánh số đến những người bạn của người đó, thì rõ ràng ta có thể thấy ý tưởng thuật toán BFS lộ rõ ra ngay từ suy nghĩ này. Như vậy, ý tưởng của ta là ta sẽ quy định những người này là một đỉnh của đồ thị, sau đó sử dụng BFS để loang ra đánh số đỉnh tương ứng. Để có thể ánh xạ được từ 1 chuỗi sang một con số để có thể đánh số được đỉnh của đồ thị, ta sử dụng một map <string, int>. Khi ta nhập một chuỗi s, nếu như map[s] = 0, đồng nghĩa với việc người có tên là s chưa được đánh số đồ thị thì ta tiến hành đánh số đỉnh cho người đó.

Độ phức tạp: $O(N \log N)$

Big-O Coding